User Manual

http://dreamteck.io

https://www.facebook.com/dreamteckstudio

Contact the team: team@dreamteck.io

Contact support: support@dreamteck.io

Discord Server: https://discord.gg/bkYDq8v

Contents

1. Introduction

Blenda is an audio editing plugin for Unity which provides an intuitive user interface and API for audio blending inside the Unity editor and during runtime. This plugin was initially created to aid the Dreamteck game studio in the development process of an original project and was then further developed to serve a more general purpose. It's core purpose is to allow for easy and designer-friendly control over sound blending. Applications include all sorts of continuous sounds like vehicle sounds, ambient sounds, adaptive music and many more.

1.1. Key Features

2. Getting Started

This chapter will guide the user through the installation, configuration and usage of Blenda.

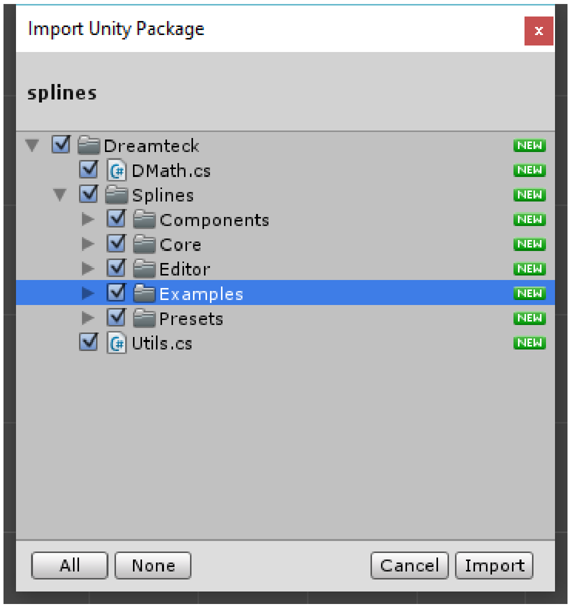
2.1. First time install

Blenda comes packed in a .unitypackage file. Upon import, all the files and directories will be listed.

The "Examples" folder is optional. It contains example scenes and scripts which demonstrate the features of the system. It can be safely excluded from the import or deleted later if it isn't needed.

Clicking "Import" will decompress and import all of the selected files in the current Unity Project.

Blenda comes with an additional PlaymakerActions.unitypackage file found inside the Dreamteck/Blenda folder. Importing this package will add Playmaker actions for use with Blenda.



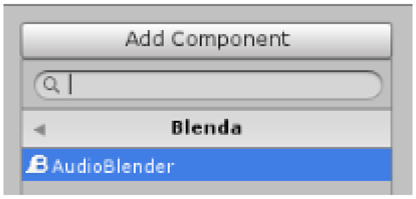
- ☑ Dreamteck NEW
  - ☑ DMath.cs NEW
  - ☑ Splines NEW
    - ☑ Components NEW
    - ☑ Core NEW
    - ☑ Editor NEW
    - ☑ Examples NEW
    - ☑ Presets NEW
  - ☑ Utils.cs NEW

2.2. Updating Blenda

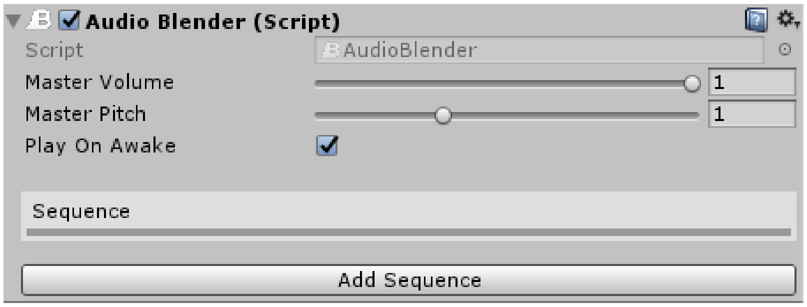
If you already have Blenda installed and have downloaded a newer version from the Asset Store, it is recommended that you delete the entire Dreamteck/Blenda folder before importing the new version.

3. The AudioBlender component

The Blenda system works through a single component – the AudioBlender component. This component is used to edit, manage and play sounds during runtime. The AudioBlender component can be added to any Game Object using the Add Component button using the path "Dreamteck/Blenda/AudioBlender".



Once added, the component will provide a "Master Volume" and a "Master Pitch" sliders for controlling the overall volume and pitch of the sound mix, as well as a sequence box and an "Add Sequence" button.

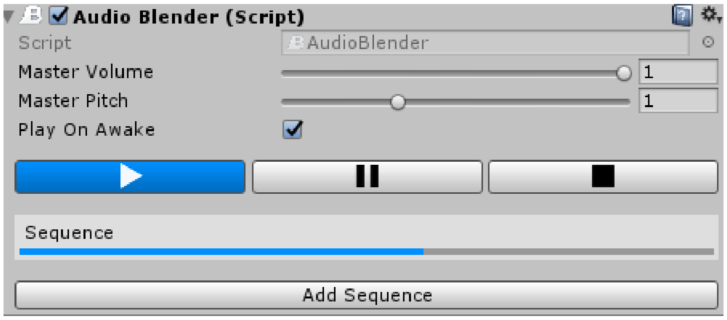


Script AudioBlender
Master Volume 1
Master Pitch 1
Play On Awake ☑

By default, the Audio Blender comes with a single empty sequence that plays on awake. An empty object called "Sequence" will also be created as a child of the Audio Blender object. Blenda organizes all of the audio sources and sequences as child objects of the Audio Blender object. As sounds are added to the sequence, more child objects are created. Deleting any child object manually from the scene, will cause it to be re-created as soon as the Audio Blender is selected or the game is played.

During Play Mode (game runtime), the Audio Blender component gets an additional set of buttons for controlling the play state of the component.

These buttons serve a debug purpose. The AudioBlender can be played, stopped and paused without the need of a script.

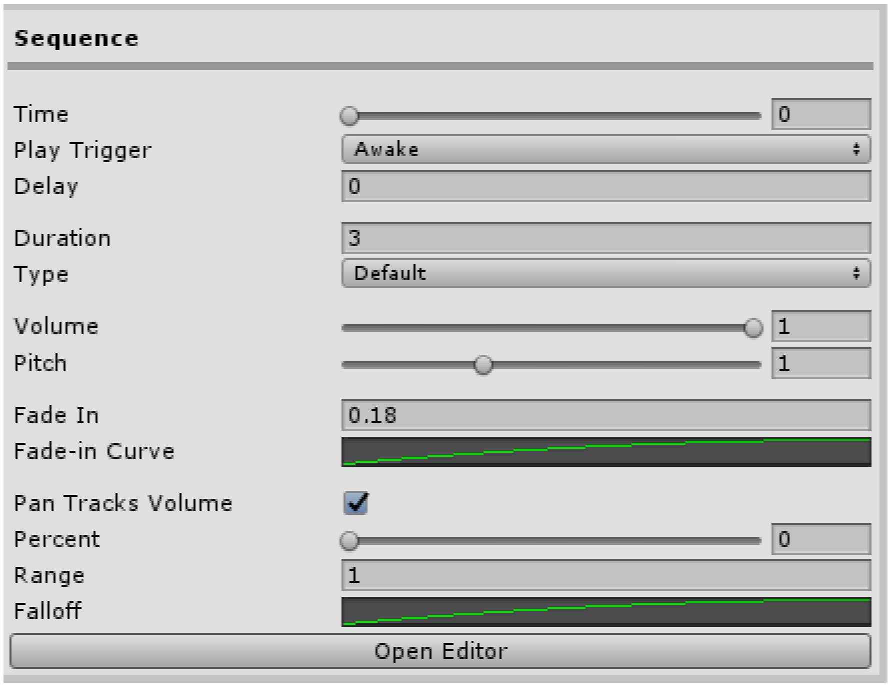


Script AudioBlender
Master Volume 1
Master Pitch 1
Play On Awake ☑

3.1. Sequences

The sequence is a collection of mixed sounds, unified by a set of properties. In order to have a sound mix, the AudioBlender needs to have at least one sequence. Sequences can be added using the “Add Sequence” button and can be deleting by right-clicking on them and selecting “Delete”. Sequences can also be renamed by right-clicking and selecting “Rename”.

Clicking on the name of a sequence will show its properties:



Sequence

Time 0
Play Trigger Awake
Delay 0
Duration 3
Type Default
Volume 1
Pitch 1
Fade In 0.18
Fade-in Curve
Pan Tracks Volume ☑
Percent 0
Range 1
Falloff

3.1.1. Sequence Properties

The properties of the sequence are:

3.1.1.1. Duration

The total duration of the sequence in seconds. There is no limit to how long a sequence can be but the duration cannot be a negative number.

3.1.1.2. Time

The current time evaluation of the sequence in seconds. The time can be anything between 0 and the duration of the sequence. The time can be modified during runtime.

3.1.1.3. Type

The Sequence type defines how time inside the Sequence is handled.

3.1.1.3.1. Default

Plays the entire sequence once and when the time reaches the end, the sequence stops playing.

3.1.1.3.2. Loop

Loops the sequence’s time a set number of times or indefinitely if the loop count is set to 0.

3.1.1.3.3. Custom

Time is not handled automatically handled by the sequence. Instead, time can be controlled by a script or set to a constant value.

3.1.1.4. Play Trigger

The Play Trigger defines when the sequence will be played. Available options are:

- Awake – Will play the sequence as soon as the object is loaded and enabled
- With Previous – Will play the sequence along with the previous sequence (if such exists)
- After Previous – Will play the sequence after the previous sequence has stopped
- None – Will not play the sequence automatically – used for manual control

3.1.1.5. Delay

An optional delay to hold the play state of the sequence. If bigger than zero, the sequence will wait the set time before playing.

3.1.1.6. Volume

A volume multiplier for the sequence. It affects all clips in the sequence's mix.

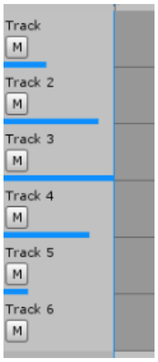
3.1.1.7. Pitch

A pitch multiplier for the sequence. It affects all clips in the sequence's mix as well as the sequence's time property if the type is set to Default or Loop.

3.1.1.8. Fade-In

A fade-in option for the volume of the sequence. If larger than zero, the sequence's volume will fade in when the sequence starts playing. Useful for blending sequences together. The effect uses a fade-in curve to define the interpolation of the volume over time.

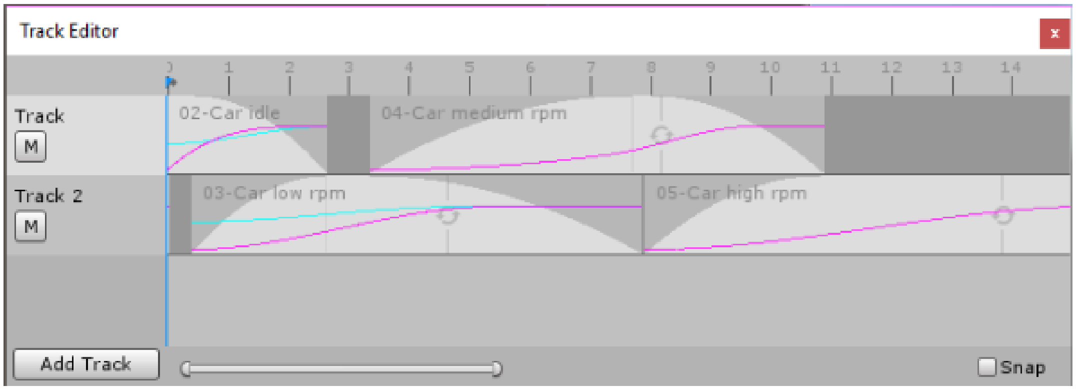
3.1.1.9. Pan Tracks Volume



If set to true, pan volume control will be enabled for the sequence's tracks. The percent slider defines which is the track with the highest volume, zero being the first track and 1 being the last track. The range property defines how many neighboring tracks will be affected and the Falloff curve defines how the volume is distributed over the range.

3.2. The Track Editor

The Track Editor can be accessed by clicking the “Open Editor” button of a sequence and it’s the user interface for blending sounds. The track editor always has the duration of the sequence it belongs to so if the sequence is 10 seconds long, then the track editor will display a 10 second timeline.



3.2.1. Managing Tracks

The tracks inside the Track Editor can hold a single, or multiple clips.

By default, the track editor comes with a single track but more tracks can be added by clicking the “Add Track” button. Having more tracks means that more clips can be played simultaneously.

Tracks can be renamed by right-clicking on their name and selecting “Rename”.

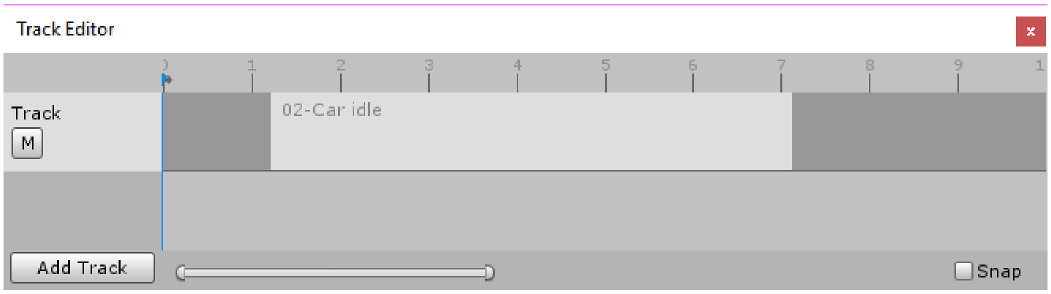
To delete a track, right-click on it and select “Delete” – this will also remove all clips contained in the track.

3.2.2. Selecting tracks

Tracks can be selected to become the current editing active track. Selection is used for duplicating clips. When a clip is copied and pasted, it will be pasted in the current active track. Any left or right click anywhere inside a track will select it. There can be only one selected track.

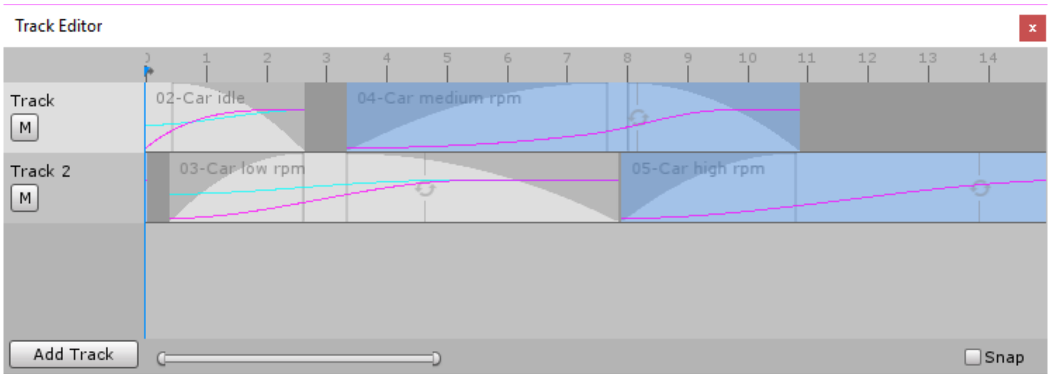
3.2.3. Adding clips

To add a clip to a track, navigate to the Audio Clip in the project manager and drag it to the track. When a clip is dragged over a track, the track will get highlighted. Dropping the clip over the track will place it at the position of the mouse. Doing so will create a child Game Object of the Audio Blender which has an Audio Source and the given clip assigned to it.



3.2.4. Selecting clips

Clips can be selected by left-clicking on them. When a clip is selected, it gets highlighted in blue.



More than one clip can be selected by holding Left-Ctrl.

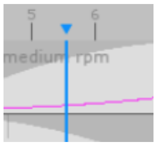
3.2.5. Removing clips

To remove a clip from a track, right-click on it and select "Delete" from the context menu. Clips can also be deleted by selecting them in the track editor and pressing the "Delete" key on the Keyboard. Doing so will also remove the created Game Object from the Audio Blender's hierarchy.

3.2.6. Duplicating clips

Clips can be duplicated. To duplicate one or more clips, select them and right-click on one of them. Choose "Copy" from the context menu, then select the desired track to paste them in, right-click inside it and choose "Paste". This will paste the copied clips inside the selected track, at the position of the tracker marker.

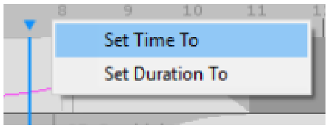
3.2.7. The Tracker Marker



The tracker marker is a blue, vertical marker that can be moved by clicking inside the time track. The tracker marker can be used to position clips and to set the time and duration of the sequence.

Right-clicking inside the time track will open a context menu with the following options:

- Set Time To
- Set Duration To



Choosing "Set Time To" will set the time of the sequence to the time of the tracker marker. Choosing "Set Duration To" will set the duration of the sequence to the time of the tracker. Warning: If there are clips inside the tracks which start after the tracker marker, setting the duration of the sequence via the tracker marker will leave them outside the bounds of the sequence and therefore they will never play.

3.2.8. Editing Clips

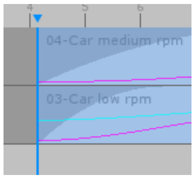
Blenda provides an intuitive way to edit clips inside the Track Editor.

3.2.8.1. Moving Clips

To move a clip, hover over it with the mouse then press and hold the left mouse button and move the mouse. This can also be done with multiple clips selected.

Except horizontally, clips can be moved vertically – between tracks. This will assign the moved clip(s) to a different track.

The tracker marker can also be used to position clips. Pressing the "[" or the "]" keys on the keyboard will snap the respective side of the currently selected clip(s) to the tracker marker.

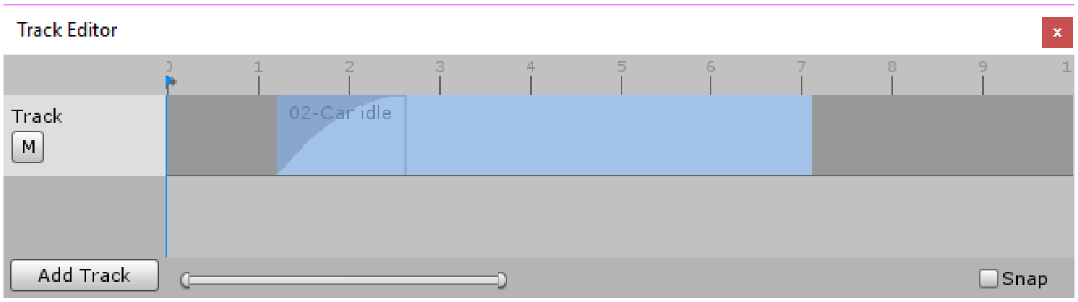


3.2.8.2. Trimming Clips

To trim a clip's length, move the mouse cursor to the left or the right edge of the clip inside the track. Doing so should change the mouse cursor to a horizontal dual arrow: ↔ When the mouse cursor is changed, clicking and dragging the mouse will trim the selected clip. This also works for multiple selected clips.

3.2.8.3. Fading

The volume of clips can be modified to fade near the beginning or the end. To control the fade inside the Track Editor, hold the Alt key, Bring the mouse cursor to the edge of a clip and drag the mouse towards the other end of the clip. Instead of trimming the clip, this will produce a curve at the given edge of the clip.



3.2.8.4. Snapping

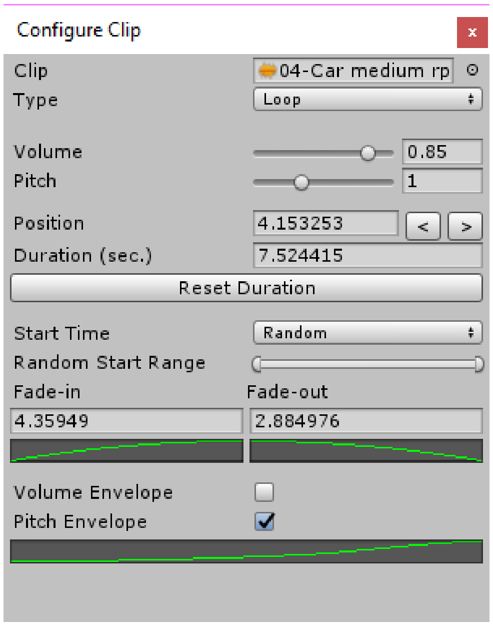
The Track Editor has a snapping feature that will snap the edges of the selected clips and the Tracker Marker to the edges of the nearest unselected clips. To turn snapping on and off, toggle the "Snap" checkbox in the bottom right corner of the Track Editor window.

3.2.9. Using the Configure Clip Window

Each clip's properties can be configured in a separate window – the Configure Clip Window. To open the Configure Clip Window, right-click on a clip inside the Track Editor and select "Settings" from the context menu.

At the top of the window there is an Audio Clip field that holds the reference to the current audio clip. The audio clip can be changed at any time – this will not change the position or duration of the clip inside the Track Editor.

Clips behave differently according to their type inside the Track Editor.



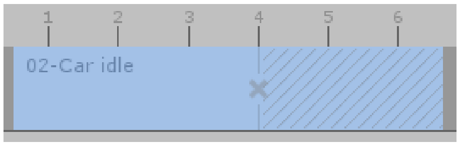
Configure Clip

Clip: 04-Car medium rp
Type: Loop
Volume: 0.85
Pitch: 1
Position: 4.153253
Duration (sec.): 7.524415

Start Time: Random
Random Start Range
Fade-in: 4.35949
Fade-out: 2.884976
Volume Envelope: □
Pitch Envelope: ☑

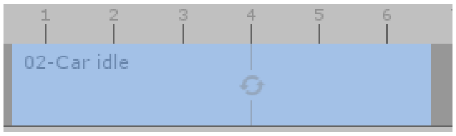
3.2.9.1. *Default Clip Type*

Plays once based on the sequence's time and the clip's duration. If the duration is shorter than the audio clip's length, the clip might not play fully. If the sequence's type is set to "Custom" and the time tracker stays over the clip, the clip will still play once and will not replay as long as the time tracker is over it. If the duration of the clip is longer than the audio clip's length, then the audio clip will not continue playing past it's end point. When the duration is longer than the audio clip's length, there will be empty space, displayed with stripes. This indicates that this part of the clip is silent – nothing will be played if the tracker falls over it. Usually having empty spaces isn't desirable.



3.2.9.2. *Loop Clip Type*

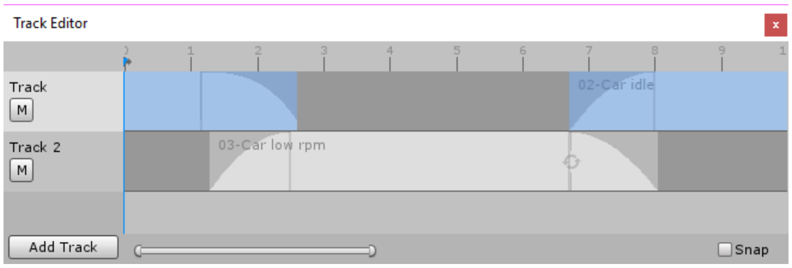
Plays continuously as long as the time tracker is over the clip. If the sequence's type is set to "Custom" and the time tracker stays over the clip, the clip will keep looping until the time tracker is no longer over it. If the duration of the clip is longer than the audio clip's length, then the audio clip will repeat as many times as the duration is longer than the clip's length. When the duration is longer than the audio clip's length, loop markers will be shown at the points where the audio clip repeats.



It is very important to be noted that the loop markers and the empty space visualization are only approximations – they are not 100% accurate especially when the sequence type is set to Custom or pitch envelopes are used.

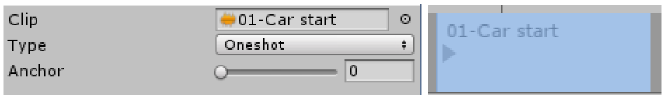
Looped clips can be positioned at the end of the sequence and they will wrap around and come back at the beginning of the sequence.

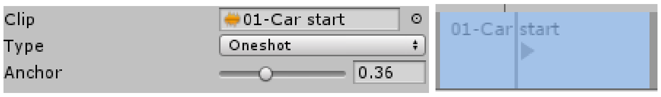
This helps to create seamless loopable sequences.



3.2.9.3. Oneshot Clip Type

Plays once from start to finish regardless of duration and time. In Oneshot mode, the clip has a trigger anchor. The trigger anchor is a percent [0-1] that defines the position along the clip that will trigger the clip to play as soon as the time tracker goes over it. The anchor is displayed inside the Track Editor.


Clip 01-Car start
Type Oneshot
Anchor 0


Clip 01-Car start
Type Oneshot
Anchor 0.36

The clip will play once regardless of how time is controlled. In Custom sequence mode, if the time tracker triggers the clip and then gets moved away from it, the clip will continue playing until it’s done.

In Oneshot mode, the clips’ durations cannot be longer then the audio clips’ length.

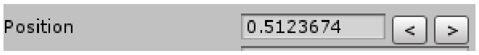
3.2.9.4. Volume & Pitch sliders

The volume and pitch sliders serve as multipliers for the volume and the pitch. This means that the Master volume of the Audio Blender, the Volume of the Sequence and the Pitch of the Sequence are multiplied by the volume and the pitch of each clip. For example, if the volume of the sequence is set to 50% and the volume of the clip is set to 50%, the final volume of the audio source will be 25% and the same applies for the pitch.

3.2.9.5. Position

The clip’s position value is exposed in the Configure Clip Window. This allows for more precise editing. Modifying the position value will move the clip inside the Track Editor.

Along with the position field, two arrow buttons are provided:


Position 0.5123674

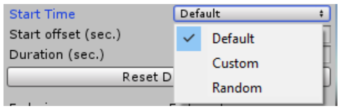
Clicking the left arrow button will position the clip at the beginning of the track and clicking the right arrow button will position the clip at the end of the track so that the clip’s right edge touches the end of the track precisely.

3.2.9.6. Duration

The duration of a clip in seconds can be set precisely using the Duration field.

Clicking the “Reset Duration” button below the Duration field will set the duration of the clip to match the audio clip’s actual length. The pitch of the clip is also taken into consideration when setting the duration. Resetting the duration will make sure that in Default or Looped sequence mode, the clip will play entirely from start to finish.

3.2.9.7. Start Time



Start Time: Default

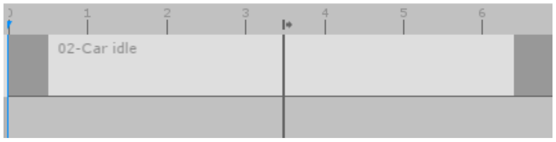
Start offset (sec.)

Duration (sec.)

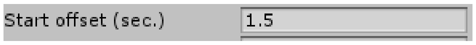
The Start Time property defines the time at which the audio clip will start playing when the time tracker falls over it.

3.2.9.7.1. Default Start Time

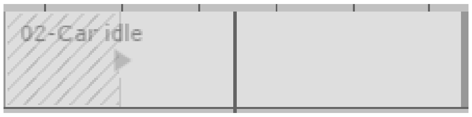
The Default start time takes the position of the time tracker into consideration. For example, if Custom type is selected for the sequence and the time tracker is suddenly moved from zero to the middle of the clip, the audio clip, associated with that clip will play from the middle:



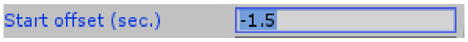
In default start time mode, the Start Offset property is also taken into account. If the offset is set to a positive number, the clip will have a forward offset, meaning that the beginning of the audio clip will be moved forward in the track editor.



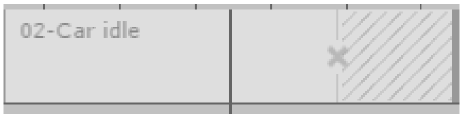
Start offset (sec.) 1.5



If the start offset is set to a negative value, then the clip’s start time will be moved backwards:



Start offset (sec.) -1.5



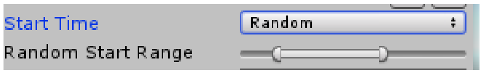
3.2.9.7.2. Custom Start Time

The custom start time disregards the position of the time track and always plays the clip from the start. If the start offset is different than zero, then the start offset will be taken into account.

3.2.9.7.3. Random Start Time

Similar to the Custom start time, the Random start time will disregard the position of the time tracker. The difference here is that instead of a fixed start time, this mode picks a random starting point within the audio clip's length.

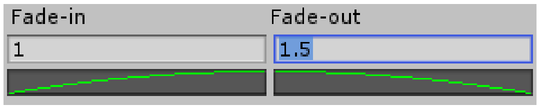
A Random Start range min-max slider is provided which can be used to define the range within which the start position can be. This range is relative to the audio clip's length, not the clip's duration inside the Track Editor. For example, if the audio clip's length is 3 seconds and the clip's duration in the Track Editor is 6 seconds, the Random Start Range will define a range between 0 and 3, not between 0 and 6.


Start Time: Random
Random Start Range

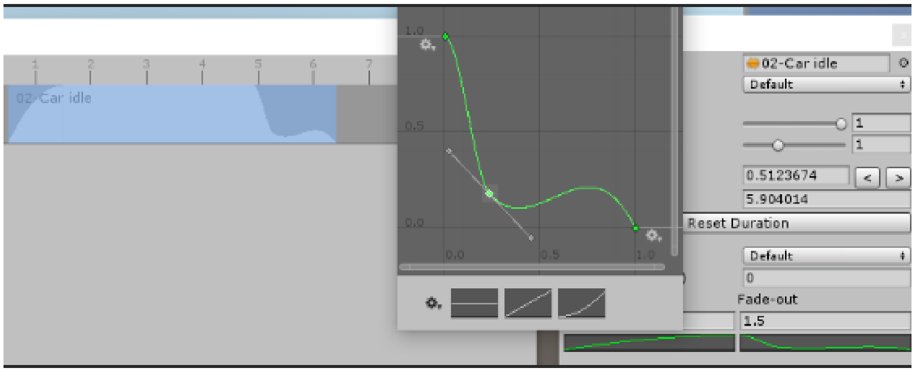
This works especially well with looped clips when variation is needed.

3.2.9.7.4. Fading

The fade-in and fade-out properties of a clip can be set through the Clip Configuration Window.

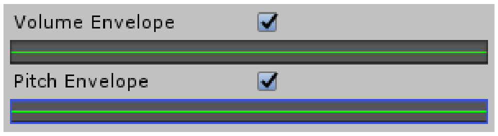

Fade-in: 1
Fade-out: 1.5

The Clip Configuration Window also exposes the interpolation curves for both fade properties. Modifying the fade curves in the configuration window will also update their appearance in the Track Editor:



02-Car idle
Default
1
1
0.5123674
5.904014
Default
0
Fade-out
1.5

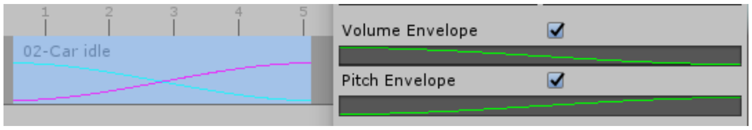
3.2.9.7.5. Envelopes


Volume Envelope ☑

Pitch Envelope ☑

The Clip Configuration Window provides two envelope features – Volume Envelope and Pitch Envelope. The envelopes are curves that control the volume or the pitch of the clip throughout its duration.

In order to use the Volume or the Pitch envelope, the respective checkbox needs to be toggled. Once toggled, a curve field will appear. It is advised to start off with a default, straight 1-1 curve. This type of curve does not apply any changes to the volume or the pitch and is a good starting point for modifications. Once a curve is present, it will be drawn in the Track Editor along the length of the clip. The Volume envelope is drawn in cyan while the pitch envelope is drawn in magenta.



Volume Envelope ☑

Pitch Envelope ☑

The volume envelope is multiplied by the Master Volume of the Audio Blender, the Volume of the Sequence and the Volume of the clip.

The pitch envelope is multiplied by the pitch of the sequence and the pitch of the clip.

4. The API

Everything that the Blenda editor does is also achievable with code during runtime. Entire sequences and mixes can be constructed and played using the Blenda API. The following chapter explores these possibilities and provides a list of all public members of the AudioBlender, Sequence, AudioTrack and BlendClip classes.

4.1. Namespaces

In order to use the Blenda API, the following namespace has to be included:

Dreamteck.Blenda	The core namespace where the AudioBlender component and all of its components are located.

This will make the AudioBlender and all of its sub-components available.

4.2. AudioBlender

The AudioBlender is a standard MonoBehaviour component which can be obtained using the GetComponent method. It contains methods for adding, removing, playing and stopping sequences.

```
using UnityEngine;
using Dreamteck.Blenda;

public class BlenderController : MonoBehaviour {
    public string sequenceName = "Sequence"; //The name of the sequence we want to
control
    private AudioBlender blender; //Reference to the AudioBlender
    private Sequence sequence; //Reference to the Sequence

       void Start () {
        blender = GetComponent<AudioBlender>(); //Get the AudioBlender component attached
to this Game Object
        sequence = blender.GetSequence(sequenceName); //Get the sequence by name
        sequence.type = Sequence.WrapMode.Custom; //Set the type to Custom so that we can
control the time manually
       }

       void Update () {
        if (sequence == null) return; //If the sequence doesn't exist, don't execute
        if (Input.GetKey(KeyCode.Space)) sequence.time = Mathf.MoveTowards(sequence.time,
sequence.duration, Time.deltaTime); //Move the time forward if Space is held
        else sequence.time = Mathf.MoveTowards(sequence.time, 0f, Time.deltaTime);
//Otherwise, move the time backward
    }
}
```

Public Properties

float masterVolume	The master volume [0-1] of the AudioBlender
float masterPitch	The master pitch [0.01-3] of the AudioBlender
bool playOnAwake	Should the AudioBlender play as soon as it is enabled?

Read-only Properties

int sequenceCount	The number of sequences this Audio Blender contains
bool isPlaying	Is the AudioBlender currently playing?
bool isPaused	Is the AudioBlender currently paused?
bool finished	Have all Sequences inside the AudioBlender finished playing?

Public Methods

void Play()	Plays the AudioBlender. The AudioBlender is enabled by default so this doesn't need to be called.
void Pause()	Pauses the AudioBlender and all of its sequences
void Stop()	Stops the AudioBlender and all of its sequences
void AddSequence(string name)	Adds a new empty sequence with a given name to the AudioBlender
void RemoveSequence(string name)	Removes a sequence with a given name from the AudioBlender – this will destroy all contained clips and tracks
void RemoveSequence(int index)	An overload which uses an index instead of name (faster)
void RemoveSequence(Sequence sequence)	An overload which uses a sequence reference
Sequence GetSequence(string name)	Returns the sequence with the given name. If no sequence with such name is found, it will return NULL
Sequence GetSequence(int index)	An overload which uses a sequence index instead of name (faster)
void PlaySequence(string name)	Plays the sequence with the given name. If no sequence with such name is found, nothing will be played
void PlaySequence(int index)	An overload which uses an index instead of name (faster)
void PauseSequence(string name)	Pauses the sequence with the given name. If no sequence with such name is found, nothing will be paused
void PauseSequence(int index)	An overload which uses an index instead of name (faster)
void StopSequence(string name)	Stops the sequence with the given name. If no sequence with such name is found, nothing will be stopped
void StopSequence(int index)	An overload which uses an index instead of name (faster)

4.3. Sequence

The Sequence is a class, nested inside the AudioBlender. Sequences should not be created outside of the AudioBlender. Use AudioBlender.AddSequence to create new sequences and AudioBlender.GetSequence to get a reference to a sequence. The properties and methods of Sequence references can be accessed directly. Modifying them will update the given Sequence inside the AudioBlender.

Enumerations

PlayTrigger {None, Awake, WithPrevious, AfterPrevious }	The event that plays the Sequence
WrapMode { Default, Loop, Custom }	The time wrap type of the Sequence. Default and Loop will manage the time automatically while Custom will allow for manual control.

Public Properties

PlayTrigger playTrigger	The event that plays the sequence
WrapMode type	The time wrap type of the Sequence
string name	The name of the Sequence
float playDelay	The Delay after which the Sequence will start playing
float volume	The volume multiplier of the Sequence
float pitch	The pitch multiplier of the Sequence
int loops	The loop count of the Sequence (applies when the type is set to WrapMode.Loop). Setting this to 0 will loop the Sequence indefinitely
float time	The current time of the Sequence in seconds
float normalizedTime	The normalized time of the Sequence [0-1]
float duration	The duration of the Sequence
float fadeIn	Fade-in for the Sequence in seconds
AnimationCurve fadeInCurve	The Fade-in interpolation curve
bool panTracks	Is track volume panning enabled?
float trackPanPercent	The track pan center percent [0-1]
AnimationCurve trackPanFalloff	The falloff curve for the track volume panning
float panRange	The pan range of the track volume pan

Read-only Properties

int trackCount	The number of tracks this Sequence contains
Transform parent	A reference to the generated transform for this

	Sequence
AudioBlender blender	A reference to the AudioBlender this Sequence belongs to
bool isDone	Is the Sequence done? Becomes true when the Sequence is done playing. Will not become true when sequence type is set to Custom or the sequence type is set to Loop and the loop count is 0 (infinite).

Public Methods

void Play()	Plays the Sequence
void Pause()	Pauses the Sequence
void Stop()	Stops the Sequence. Sequences after it with the “After Previous” play trigger will not play
void Finish()	Stops the Sequence but also marks it as done so that the next Sequence can start playing
AudioTrack AddTrack(string name)	Adds a track with a given name to the Sequence and returns a reference to it
AudioTrack[] AddTracks(int count)	Bulk-adds multiple tracks to the Sequence and returns an array with references to them
void RemoveTrack(string name)	Removes the track with the given name from the Sequence
void RemoveTrack(int index)	An overload which uses a track index instead of name (faster)
void RemoveTrack(AudioTrack track)	An overload which uses a track reference instead of name
AudioTrack GetTrack(int index)	Gets the track with the given index
AudioTrack GetTrack(string name)	Gets the track with the given name

4.4. AudioTrack

Similar to the Sequence, the AudioTrack is a class, nested inside the Sequence and instances should not be created outside of the Sequence class. AudioTracks can be created using the Sequence.AddTrack method.

Public Properties

string name	The name of the AudioTrack
bool mute	Is the AudioTrack muted?

Read-only Properties

int clipCount	The number of clips this AudioTrack contains
Sequence sequence	The sequence this track belongs to

Public Methods

void AddClip(AudioClip clip, float time)	Adds a new clip to the track at the given time
void AddClip(BlendClip clip)	Adds a new BlendClip to the track. Used when moving a clip from one track to another. Works between Sequences
void RemoveClip(int index)	Removes and destroys a clip by index
void RemoveClip(BlendClip clip)	Removes and destroys a clip by reference
void RemoveClip(AudioClip clip)	Removes and destroys a clip by reference to its AudioClip
void UnlinkClip(int index)	Removes a clip by index without destroying its AudioSource
void UnlinkClip(BlendClip clip)	Removes a clip by reference without destroying its AudioSource
void UnlinkClip(AudioClip clip)	Removes a clip by reference to its AudioClip without destroying its AudioSource
BlendClip GetClip(int index)	Gets a clip by index
BlendClip GetClip(AudioClip audioClip)	Gets a clip by reference to its AudioClip

4.5. BlendClip

The BlendClip is nested inside the AudioTrack class and it’s a wrapper for AudioClip objects. This is the class that the Track Editor manipulates in the User Interface.

Enumerations

PlayType { Default, Loop, Oneshot }	The behavior type of this clip
StartTimeMode { Default, Custom, Random }	Defines how the clip’s time is handled when it starts playing

Public Properties

AudioClip audioClip	The AudioClip of this blend clip
float start	The clip’s starting position along the timeline
float duration	The clip’s duration in the timeline
float fadeIn	The clip’s fade-in duration
float fadeOut	The clip’s fade-out duration
AnimationCurve fadeInCurve	The fade-in curve which controls the fade-in interpolation
AnimationCurve fadeOutCurve	The fade-out curve which controls the fade-out interpolation
float volume	The clip’s volume multiplier [0-1]
float pitch	The clip’s pitch multiplier
PlayType playType	The clip’s type
StartTimeMode startTimeMode	The start time of the clip
float startOffset	The start offset of the clip
float minRandomStartTime	The minimal normalized time range when Random start time is selected
float maxRandomStartTime	The maximal normalized time range when Random start time is selected
bool usePitchEnvelope	Should the clip use the pitch envelope?
bool useVolumeEnvelope	Should the clip use the volume envelope?
AnimationCurve pitchEnvelope	The pitch envelope curve of the clip
AnimationCurve volumeEnvelope	The volume envelope curve of the clip
float oneShotAnchor	The trigger anchor for Oneshot clips [0-1]

Read-only Properties

GameObject gameObject	The game object for the created Audio Source for this clip
AudioTrack track	The AudioTrack this clip belongs to
float end	The end position of the clip, equal to start + duration.

Public Methods

BlendClip Copy()	Duplicates the clip with all of its properties.